

## Avant-propos

<b>1. Une transformation digitale</b>	<b>13</b>
<b>2. Pourquoi ce livre ?</b>	<b>14</b>
<b>3. À qui est destiné ce livre ?</b>	<b>14</b>
3.1 Cas n°1 - C'était mieux avant !	14
3.2 Cas n°2 - Comment puis-je améliorer mon agilité dans mes tests ?	15
3.3 Cas n°3 - Mon équipe veut passer au DevOps, quels sont mes besoins de tests ?	15
3.4 Cas n°4 - Je suis dans une équipe agile en « Fast IT » et tout le monde code et teste	16
<b>4. Comment lire ce livre ?</b>	<b>16</b>
<b>5. Qui suis-je ?</b>	<b>18</b>

## Quelques rappels

<b>1. Introduction</b>	<b>21</b>
<b>2. Les 4 valeurs de l'agile - Rappel du Manifeste Agile</b>	<b>22</b>
<b>3. Les 7 principes du test - Rappel ISTQB</b>	<b>23</b>

3.1 Les tests montrent la présence de défauts	23
3.2 Les tests exhaustifs sont impossibles	24
3.3 Tester tôt	24
3.4 Regroupement des défauts	26
3.5 Paradoxe du pesticide	26
3.6 Les tests dépendent du contexte	28
3.7 Illusion de l'absence de défaut	28
<b>4. Techniques de test de base</b>	<b>29</b>
4.1 Partitionnement de classes d'équivalences et valeurs limites	29
4.2 Heuristique CORRECT	32
4.3 Table de décision	34
4.3.1 Approche analytique	34
4.3.2 Heuristique - Test n-wise	36
4.4 Transitions d'états	37
<b>5. Le Cycle de vie logiciel</b>	<b>39</b>
5.1 Projet à l'échelle d'une seule équipe SCRUM	39
5.1.1 Scrum en quelques lignes	40
5.1.2 Sprint Refinement Meeting (grooming)	42
5.1.3 Sprint planning	44
5.1.4 Daily Scrum Meeting	

5.1.5 Sprint Review	47
5.1.6 Rétrospective	50
5.1.7 Réflexions sur les rétrospectives	51
5.2 Projet à l'échelle de plusieurs équipes	63
5.2.1 Cas de frameworks agiles existants	65
5.2.2 Point de vue du modèle SAFe	66
5.2.3 Point de vue du modèle Spotify	66
5.2.4 Point de vue Less	72
5.2.5 Point de vue DAD - « Disciplined Agile Delivery »	76
5.2.6 Point de vue Scrum@Scale - S@S	79
5.3 Approche Composant vs Fonctionnalité	82
	84
L'état d'esprit	
<b>1. Introduction - Exemple de l'industrie du textile</b>	<b>91</b>
1.1 Contrôle des ballots	93
1.2 Contrôle pendant le tissage	93
1.3 Amélioration du procédé de tissage	94
1.4 Le test agile	95
<b>2. État d'esprit Lean</b>	<b>95</b>

2.1 Le Lean, les origines de l'agile	95
2.2 Quelques pratiques Lean utiles au test	96
2.2.1 Kanban	97
2.2.2 Juste à Temps - « Just in time »	97
2.2.3 5S	98
2.2.4 Intégrer la qualité dès la conception	99
2.2.5 Le First Pass Yield : faire bon du 1er coup !	100
2.2.6 Analyse de cause racine	100
2.2.7 Petits stocks, régularité des flux	109
2.3 Manifeste du test agile	109
2.3.1 Tester pendant le sprint	110
2.3.2 Prévenir les bugs	111
2.3.3 Comprendre ce que l'on doit tester	113
2.3.4 Concevoir un meilleur système	114
2.3.5 L'équipe est responsable de la qualité du produit	114

## Versant métier

<b>1. Introduction</b>	<b>125</b>
<b>2. Définition du Prêt (DoR)</b>	<b>125</b>

<b>3. Clarté d'une US</b>	<b>129</b>
<b>4. Test des US</b>	<b>130</b>
4.1 Ingénierie des critères d'acceptation des US	130
4.1.1 Définition du Terminé (DoD)	131
4.1.2 BDD, Gherkin et ATDD	132
4.2 Préparation de la revue d'US ou d'Epic	141
4.3 Revue de définition des US	142
4.3.1 Affinement des US	142
4.3.2 Approche « Los Très Amigos »	143
4.3.3 Visite guidée	144
4.4 Revue de réalisation de l'US	145
<b>5. Le Produit</b>	<b>146</b>
<b>6. La valeur métier</b>	<b>149</b>
6.1 Modèle MoSCoW	149
6.2 Diagramme de Kano	149
6.3 Priorisations complexes (Weighted Shortest Job First - WSJF)	152
<b>7. Production des spécifications</b>	<b>153</b>
7.1 Cas des exigences et spécifications	153

7.2 Cas des US	154
7.2.1 Le rôle de chacun	154
7.2.2 Le récit utilisateur	155
7.3 Cas des Epics	157
7.3.1 Les Features - les Fonctionnalités	158
7.3.2 Les Capabilities	160
7.3.3 Les Epics du Portfolio	161

## Versant industriel du test

<b>1. Introduction</b>	<b>163</b>
<b>2. Stratégie de test</b>	<b>163</b>
2.1 Vision académique de la stratégie de test	163
2.2 Catégories d'approches	166
2.3 Tester tôt	169
2.4 Approche par le risque	171
2.4.1 Caractérisation du niveau de risque	171
2.4.2 Principe du Risk-Based Testing - RBT	174
2.4.3 Granularité	176
2.4.4 Niveau « données de tests »	176
2.4.5 Niveau « scénarios de tests »	176

2.4.6 Niveau « exigence de tests »	176
2.4.7 Niveau « campagnes de tests »	177
2.4.8 Exemple de gestion des campagnes de tests de régression	177
2.4.9 Périmètre vs Qualité et objectifs de test	179
2.5 Heuristique pour une stratégie de test	180
2.6 Context-Driven Testing (CDT) - Test Piloté par le Contexte	182
2.6.1 Valeur d'une pratique dépendante du contexte	186
2.6.2 Les Best Practices	187
2.6.3 Les gens qui travaillent ensemble	188
2.6.4 Prédicibilité des projets	188
2.6.5 Résolution du problème	189
2.6.6 Le challenge intellectuel	189
2.6.7 Esprit critique	189
<b>3. Tests systèmes et bout-en-bout</b>	<b>189</b>
3.1 Organisation des tests scénarisés	190
3.2 Ingénierie des Tests systèmes/bout-en-bout	191
3.3 Campagne de test	192
3.3.1 Planification	193
3.3.2 Monitoring et Contrôle	194
3.3.3 Clôture	194

3.4 Tests de régression	195
3.4.1 Smoke tests	195
3.4.2 Campagnes prédéfinies par le risque	195
3.4.3 Campagnes prédéfinies par thème	196
3.4.4 Check-list	198
<b>4. Tests exploratoires</b>	<b>200</b>
4.1 Préparation des tests exploratoires	200
4.1.1 La charte	202
4.1.2 La zone de recherche	202
4.1.3 L'état d'esprit - Technique de la Persona	203
4.1.4 Les moyens d'une session de TE	206
4.1.5 Session de TE avec Intervenants Externes	206
4.2 Session de Test exploratoire	208
4.3 Fin de Session de Test exploratoire	210
<b>5. Ingénierie des Tests de recette</b>	<b>210</b>
5.1 Préparation de la recette	210
5.2 Cas des certifications et des tests réglementaires	211
<b>6. Stratégies de gestion des anomalies</b>	<b>214</b>



6.1 Loi de Pareto	214
6.2 Taxonomie des erreurs	215
6.2.1 Quelques chiffres de l'industrie	215
6.2.2 Bugs ou parasites ?	216
6.2.3 Quels types de bugs ?	217
6.3 La dette technique	219
6.3.1 Mesurer la dette	220
6.3.2 Attitudes face à la dette	228
<b>7. Rôles</b>	<b>230</b>
<b>8. Reporting</b>	<b>231</b>
8.1 Généralités	231
8.2 Contenu d'un rapport	231
<b>9. Stratégies Données et Environnements de tests</b>	<b>232</b>
9.1 Clonage des données de production	233
9.2 Gestion des données de test	233
9.2.1 Génération des données	233
9.2.2 Optimisation des tests par les valeurs	234
9.2.3 Génération aléatoire des valeurs nominales	235
9.2.4 Approche combinatoire aléatoire	236

9.2.5 Generative Adversarial Network - GAN	236
9.2.6 Minimisation des données	237
9.2.7 Anonymisation des données	238
9.2.8 Cas des données sujettes à la péremption	239
9.2.9 Dictionnaire de données	239
9.2.10 Isolation des données	240
9.2.11 Synthèse	241
9.3 Reprise de données	243

## Tests non fonctionnels

<b>1. Introduction</b>	<b>247</b>
<b>2. Tests de sécurité</b>	<b>248</b>
2.1 Généralités sur les tests de sécurité	249
2.2 Orientations en termes de sécurité	251
2.3 Principes	253
2.3.1 Séparation	253
2.3.2 Contrôle	254
2.3.3 Étapes pour définir un test de sécurité	254
2.3.4 Technique de tests pour les OpSec	257
2.3.5 Types d'anomalies	

2.3.6 Activités à contrôler	262
2.4 Organisation des tests de sécurité	266
2.4.1 Modèle par jalons	268
2.4.2 Correspondance entre sécurité et activités compatibles agile	269
2.4.3 Vision SEAP	269
2.4.4 Approche par le risque	270
2.4.5 Modèle Microsoft SDL	270
2.4.6 Modèle PASTA	272
2.4.7 Vision Coveros	273
2.4.8 Security Cards	274
2.5 Quelques techniques de tests	275
2.5.1 Analyse de sécurité pour les exigences	277
2.5.2 Analyse de failles de sécurité sur le design et le code	277
2.5.3 Tests de pénétration	284
2.5.4 Détection d'intrusion	298
2.5.5 Intégration de la sécurité dans une approche DevOps	305
2.5.6 Analyse de patch & applications sur la production	309
<b>3. Tests d'utilisabilité</b>	<b>310</b>
3.1 Le test du Magicien d'Oz	311
3.2 L'Experience Map	312

3.3 Retours des utilisateurs en production	312
3.4 Accessibilité	315
3.4.1 Intérêt de l'accessibilité	316
3.4.2 Comprendre les recommandations	316
3.4.3 Certification et pénalités	320
<b>4. Tests de charges</b>	<b>321</b>
<b>5. Tests de migration ou reprise de données</b>	<b>322</b>
5.1 Généralités sur les Migrations	326
5.2 Migrations impliquant la reprise des données	329
5.2.1 Classification des données	329
5.2.2 Migration de type Big Bang	331
5.2.3 Migration au fil de l'eau	331
5.2.4 Outillage de la migration	332
5.3 Phase finale de la migration - le décommissionnement	334
<b>6. Tests de tolérance aux pannes</b>	<b>335</b>
6.1 Facteurs de défaillance	336
6.1.1 L'Information	336
6.1.2 La disponibilité	337
6.1.3 La compensation	340

6.1.4 La stabilisation	343
6.2 Conception	343
6.2.1 N+1	344
6.2.2 Fail fast	344
6.2.3 Paxos	344
6.2.4 Raft	346
6.3 Reprise de fonctionnement	347
6.3.1 Définition des tests de reprise de fonctionnement	348
6.3.2 Préparation des tests de reprise de fonctionnement	348
6.3.3 Exécution des tests de reprise de fonctionnement	349
<b>7. Tests d'exploitabilité</b>	<b>350</b>
<b>8. Tests de traçabilité</b>	<b>351</b>
8.1 Périmètre	351
8.2 La vérification	354
<b>9. Tests de sûreté de fonctionnement</b>	<b>354</b>

## Versant technique du test

### 1. Introduction

	<b>359</b>
<b>2. Test de conception et de code</b>	<b>359</b>
2.1 Préparation des tests pour la conception et le code	359
2.1.1 Responsabilité collective	360
2.1.2 Préparation de la revue de conception et de code	361
2.1.3 Tests Unitaires	377
2.1.4 Pair programming	404
2.2 Revue de conception et de code	406
2.2.1 Inspection	406
2.2.2 Méthode B	408
2.3 Software Craftsmanship	410
<b>3. Ingénierie des tests d'intégration</b>	<b>414</b>
3.1 Approche Big Bang	416
3.2 Approche Top-down	417
3.3 Approche Bottom-up	419
3.4 Approche Sandwich	421
3.5 Autres approches	422
3.5.1 Intégration par Collaboration	422
3.5.2 Intégration au Backbone	422
3.5.3 Intégration à fréquence élevée	422

3.5.4 Intégration de Services Distribués	422
3.5.5 Intégration Continue	424
<b>4. Stratégies d'automatisation</b>	<b>425</b>
4.1 Difficultés d'automatisation des tests	426
4.2 Plateformes techniques de test	426
4.2.1 Application web	427
4.2.2 Automatisation des tests pour applications web	428
4.2.3 Automatisation des tests pour navigateurs graphiques pour ordinateur personnel	429
4.2.4 Automatisation des tests pour navigateurs textuels	430
4.2.5 Application mobile	431
4.2.6 Des plateformes au secours du problème de portabilité	436
4.2.7 Application native pour ordinateur personnel	439
4.3 Le cornet de glace des tests automatisés	440
4.4 Élaboration des scripts de tests	441
4.4.1 Automatisation des tests de composants	442
4.4.2 Automatisation des tests d'intégration	443
4.4.3 Automatisation des tests fonctionnels	443
4.4.4 Compétences de programmation	444
4.4.5 Approche par mots-clés	446
4.4.6 Model-Based Testing - MBT	448

4.4.7 Priorisation pour l'automatisation	452
4.4.8 Qui automatise ?	454
4.5 Répétabilité des scripts	456
4.6 Robustesse des scripts	457
4.6.1 Concept de Jidoka	457
4.6.2 Insensibilité aux bruits - Objectif du script	458
4.6.3 Les Faux-Positifs	458
4.7 Maintenabilité des scripts	459
4.7.1 Design pattern Page Object	461
4.7.2 Objets métier	462
4.7.3 Design pattern ScreenPlay	463
4.7.4 Framework des tests automatiques	464
4.8 Exécution des scripts	465
4.8.1 Exécution en aveugle	466
4.8.2 Reporting d'exécution	466
<b>5. Environnement technique du test</b>	<b>467</b>
5.1 Alpha Testing et Beta Testing	467
5.2 Projet DevOps	469
5.2.1 Culture et généralités	469
5.2.2 Processus d'Intégration Continue (PIC)	474



5.2.3 Observabilité	490
5.2.4 Patterns & antipatterns organisationnels	490
5.2.5 Feature Flipping	495
5.2.6 Notion de Conteneur	497
5.2.7 Stratégie Shift Right	506
5.2.8 Outils	511

## Facteurs de succès

<b>1. Introduction</b>	<b>513</b>
<b>2. Le Contexte organisationnel</b>	<b>513</b>
2.1 Le Management 3.0	515
2.2 Holacratie	518
2.3 Et quoi d'autre ?	518
<b>3. Vue globalo-locale</b>	<b>521</b>
3.1 Point de vue de la régression	521
3.2 Lien entre les anomalies	522
3.3 Point de vue de l'humeur	522
3.4 En résumé	523
<b>4. Culture générale</b>	

	<b>524</b>
4.1 Gestion de la connaissance comme enjeux du test	524
4.2 Vouloir apprendre	526
4.3 Repérage de la connaissance	528
4.4 Actualisation de la connaissance	529
4.5 Préservation de la connaissance	529
4.5.1 La Solution Intent	530
4.5.2 La CMDB d'ITIL	531
4.5.3 Généralités sur les outils	532
4.6 Valorisation de la connaissance	536
4.6.1 Mise en avant d'un document	536
4.6.2 Mesure de la compétence acquise	536
4.7 Partage de la connaissance	539
4.7.1 Généralités sur la documentation	540
4.7.2 Modélisation Agile	542
4.7.3 État d'esprit du modélisateur Agile	551
4.7.4 La notion de « Ba - »	551
<b>5. Ce qui fait l'âme du Testeur</b>	<b>554</b>
5.1 Ceux qui testent	554
5.1.1 Le Pragmatique	556
5.1.2 Le Joueur	

5.1.3 Le Méthodique	556
5.1.4 L'Empirique	557
5.1.5 Le relationnel	557
5.1.6 Changement des mentalités	557
5.2 Passion et motivation intrinsèque	561
5.3 La confiance	566
5.3.1 La confiance en une version	567
5.3.2 La confiance comme attribut du Leadership	567
5.4 Biais cognitifs et test	568
5.4.1 L'intuition du Testeur	569
5.4.2 Biais de confirmation	571
5.4.3 Dissonance cognitive chez le Testeur	572
5.4.4 Effet de cadrage	573
5.4.5 Corrélation illusoire	574
5.4.6 Raisonnement fallacieux	575
5.4.7 Illusions cognitives	575
5.4.8 Identification de motifs à partir du vide	577
5.5 Pensée critique	579
5.6 Créativité	585
5.7 Le Leadership	589
5.8 Les valeurs du Testeur Agile	592

	594
Annexes	
<b>Bibliographie</b>	<b>597</b>
<b>Index</b>	<b>629</b>